This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1-30. (Canceled)

31. (Previously presented) A program storage device readable by a computer, the program storage device tangibly embodying one or more programs of instructions executable by the computer to perform a method for providing a two-step communication scheme, the method comprising:

establishing a portion of memory configured to provide asynchronous, connectionless inter-process communication between a first process and a second processes;

granting exclusive read and write access to a first process to the portion of memory configured to provide asynchronous, connectionless inter-process communication between the first process and the second process;

while having been granted to the exclusive read and write access to the portion of memory configured to provide asynchronous, connectionless inter-process communication, accessing independently of any connection to said second process the portion of memory configured to provide asynchronous, connectionless inter-process communication by the first process to modify the contents thereof to provide a message for processing by the second process; and

releasing exclusive read and write access by the first process to the portion of memory configured to provide asynchronous, connectionless inter-process communication to prevent inter-process communication between the first and second process from becoming a performance bottleneck by releasing resources of the first process after the first process modifies the contents of the portion of memory.

32. (Previously presented) The program storage device of claim 31 wherein the method further comprises configuring the memory to provide header having an operation code and a parameter region interpreted according to the operation code.

33. (Previously Presented) The program storage device of claim 31, wherein the providing the message into the portion of memory by the first process further comprises initiating a remote procedure call.

34. (Previously presented) The program storage device of claim 31 wherein the method further comprises granting exclusive read and write access to the second process to the portion of memory configured to provide asynchronous, connectionless inter-process communication, while having been granted to the exclusive read and write access to the portion of memory, accessing independently of any connection to said first process the portion of memory by the second process to access the message provided in the portion of memory by the first process and releasing exclusive read and write access by the second process to the portion of memory.

35. (Previously presented) The program storage device of claim 34 wherein the method further comprises:

establishing exclusive read and write access to the portion of memory by the second process;

accessing independently of any connection to said first process the portion of memory by the second process to provide a result message in response to the message placed in the portion of memory by the first process;

releasing exclusive read and write access by the second process to the portion of memory; and

providing by the second process a notification to the first process to check the portion of memory.

36. (Previously presented) The program storage device of claim 31 wherein the method further comprises providing by the first process a notification to the second process to check the portion of memory.

37. (Previously presented) A server comprising a memory, wherein a portion of the memory is configured to provide two-step, asynchronous, connectionless inter-process communication between a first process and a second process, the portion of memory being configured as memory accessible by the first and second processes by selective granting, wherein read and write access to the portion of memory being granted exclusively to the first process for modification of contents of the portion of memory independently of any connection to said second process to prevent inter-process communication between the first and second process from becoming a performance bottleneck by releasing resources of the first process after the first process modifies the contents of the portion of memory.

38. (Previously Presented) The server of claim 37, wherein the portion of memory comprises a slot having a header comprising an operation code and a parameter region interpreted according to the operation code.

39. (Previously Presented) The server of claim 37, wherein the placing message into the portion of memory by the first process further comprises initiating a remote procedure call.

40. (previously presented) The server of claim 37, wherein the first process releases exclusive read and write access by the first process to the portion of memory, the second process is granted exclusive read and write access to the portion of memory configured to provide asynchronous, connectionless inter-process communication, accesses the portion of memory independently of any connection to said first process to access a message provided in the portion of memory by the first process and releases exclusive read and write access by the second process to the portion of memory.

41. (Previously presented) The server of claim 40, wherein the second process is granted exclusive read and write access to the portion of memory, accesses the portion of memory independently of any connection to said first process to provide a result message in response to the message placed in the portion of memory by the first process, releases exclusive read and write access by the second process to the portion of memory and provides a notification to the first process to check the portion of memory.

42. (Previously Presented) The server of claim 37, wherein the first process provides a notification to the second process to check the portion of memory.

43. (Previously presented) A system, comprising:

a first process;

a second process; and

memory configured to provide asynchronous, interprocess communication between the first process and the second process, wherein the memory provides a portion of memory configured to be accessible by the first and second processes by selective granting, wherein read and write access to the portion of memory is granted exclusively to the first process for modification of contents of the portion of memory to prevent inter-process communication between the first and second process from becoming a performance bottleneck by releasing resources of the first process after the first process modifies the contents of the portion of memory.

44. (Previously Presented) The system of claim 43, wherein the portion of memory comprises a slot having a header comprising an operation code and a parameter region interpreted according to the operation code.

45. (Previously Presented) The system of claim 43, wherein the placing message into the portion of memory by the first process further comprises initiating a remote procedure call.

46. (previously presented) The system of claim 43, wherein the first process releases exclusive read and write access by the first process to the portion of memory, the second process is granted exclusive read and write access to the portion of memory configured to provide asynchronous, connectionless inter-process communication, accesses the portion of memory to access a message provided in the portion of memory by the first process and releases exclusive read and write access by the second process to the portion of memory.

47. (Previously presented) The system of claim 46, wherein the second process is granted exclusive read and write access to the portion of memory, accesses the portion of memory to provide a result message in response to the message placed in the portion of memory by the first process, releases exclusive read and write access by the second process to the portion of memory and provides a notification to the first process to check the portion of memory.

48. (Previously Presented) The system of claim 43, wherein the first process provides a notification to the second process to check the portion of memory.

49. (Previously presented) A service level agreement (SLA) server, comprising:

a plurality of processes, the plurality of processes comprising a database manager for managing performance data, an application server for collecting performance data and providing a client interface for establishing service level agreements, a SLA core for analyzing data and controlling actions based on service level agreements and policy and a performance monitor daemon for communicating with remote I/O service gateways to collect data and send throttling requests; and

memory configured to provide asynchronous, interprocess communication between the processes, wherein the memory provides a portion of memory configured to be accessible by the processes by selective granting, wherein read and write access to the portion of memory is granted exclusively to a first of the processes for modification of contents of the portion of memory to prevent inter-process communication between the process from becoming a performance bottleneck by releasing resources of the first of the processes after the first of the processes modifies the contents of the portion of memory.

50. (Previously presented) A service level agreement (SLA) server, comprising:

a processor configured for providing a plurality of processes; and

memory configured to provide asynchronous, interprocess communication between the first process and the second process, wherein the memory provides a portion of memory configured to be accessible by the first and second processes by selective granting, wherein the processor grants exclusive read and write access to the portion of memory by the first process for modification of contents of the portion of memory to prevent interprocess communication between the first and second process from becoming a performance bottleneck by releasing resources of the first process after the first process modifies the contents of the portion of memory.

51. (Previously presented) A method for providing a two-step communication scheme, comprising:

establishing a portion of memory configured to provide asynchronous, connectionless inter-process communication between a first process and a second processes;

granting exclusive read and write access to a first process to the portion of memory configured to provide asynchronous, connectionless inter-process communication between the first process and the second process;

while having been granted to the exclusive read and write access to the portion of memory configured to provide asynchronous, connectionless inter-process communication, accessing independently of any connection to said second process the portion of memory configured to provide asynchronous, connectionless inter-process communication by the first process to modify the contents thereof to provide a message for processing by the second process; and

releasing exclusive read and write access by the first process to the portion of memory configured to provide asynchronous, connectionless inter-process communication to prevent inter-process communication between the first and second process from becoming a performance bottleneck by releasing resources of the first process after the first process modifies the contents of the portion of memory.

52. (Previously presented) A server comprising a means for storing data, wherein a portion of the means for storing data is configured to provide two-step, asynchronous, connectionless inter-process communication between a first process and a second process, the portion of the means for storing data is configured to be accessible by the first and second processes by selective granting, wherein read and write access to the portion of the means for storing data is granted exclusively to the first process for modification of contents of the portion of the means for storing data independently of any connection to said second process to prevent inter-process communication between the first and second process from becoming a performance bottleneck by releasing resources of the first process after the first process modifies the contents of the portion of the means for storing data.

53. (Previously presented) A system, comprising:

first process means;

second process means;

means for storing data configured to provide asynchronous, interprocess communication between the first process means and the second process means, wherein the means for storing data is configured to be accessible by the first and second process means by selective granting, wherein read and write access to the portion of the means for storing data is granted exclusively to the first process means for modification of contents of the portion of the means for storing data to prevent inter-process communication between the first and second process means from becoming a performance bottleneck by releasing resources of the first process means after the first process means modifies the contents of the portion of the means for storing data.

54. (Previously presented) A service level agreement (SLA) server, comprising:

a plurality of process means, the plurality of processes comprising process means for managing performance data, process means for collecting performance data and providing a client interface for establishing service level agreements, process means for analyzing data and controlling actions based on service level agreements and policy and process means for communicating with remote I/O service gateways to collect data and send throttling requests; and

means for storing data configured to provide asynchronous, interprocess communication between the plurality of process means, wherein the means for storing data is accessible by the plurality of process means by selective granting, wherein read and write access to the means for storing data is granted exclusively to a first of the process means for modification of contents of the means for storing data to prevent inter-process communication between the plurality of process means from becoming a performance bottleneck by releasing resources of the first of the process means after the first of the process means modifies the contents of the portion of means for storing data.